**Technology Review – Classification & Customer Behavior Analysis using Sklearn and Amazon SageMaker**

In today’s E-Commerce, classifier algorithms are very popular as they can be used to classify sentiments of reviews based on words. In this Technology Review, we analyze systems that perform the classification of unstructured text data such as customer reviews by means of sentiment analysis techniques. We elaborate on the process and technologies available to extract a domain specific lexicon of semantically relevant words based on a given corpus, continuously re-train and predict classification at production scale. The resulting lexicon backs the sentiment analysis for generating a classification of the unstructured text data.

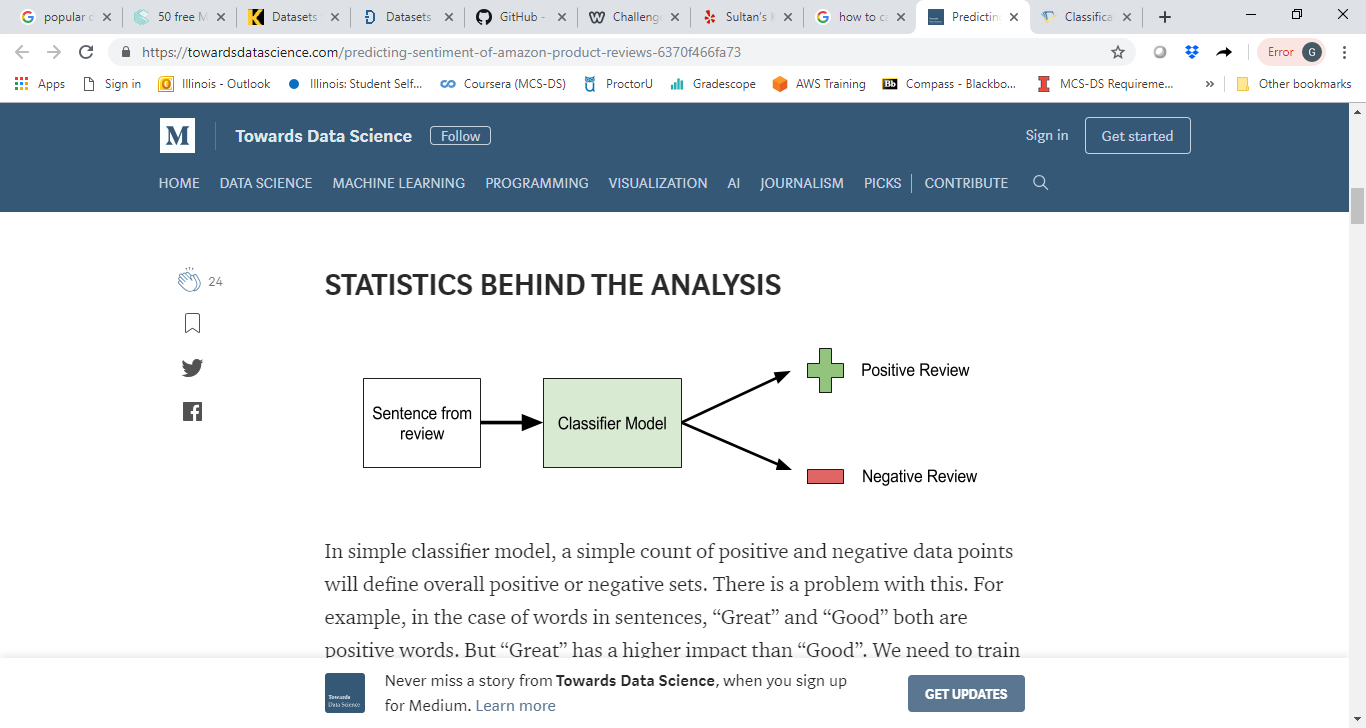
1. **Classification – Sentiment Analysis Techniques**

In machine learning, classification is used to classify a new observation into a specific set/category based on a training set of data containing observations whose category is known in advance. The most common example is “spam” or “non-spam” classes for emails. In E-Commerce, classifier algorithms can be used to classify sentiments of review based on words. The specific words in the language are categorized in advance for their positive or negative sentiments.

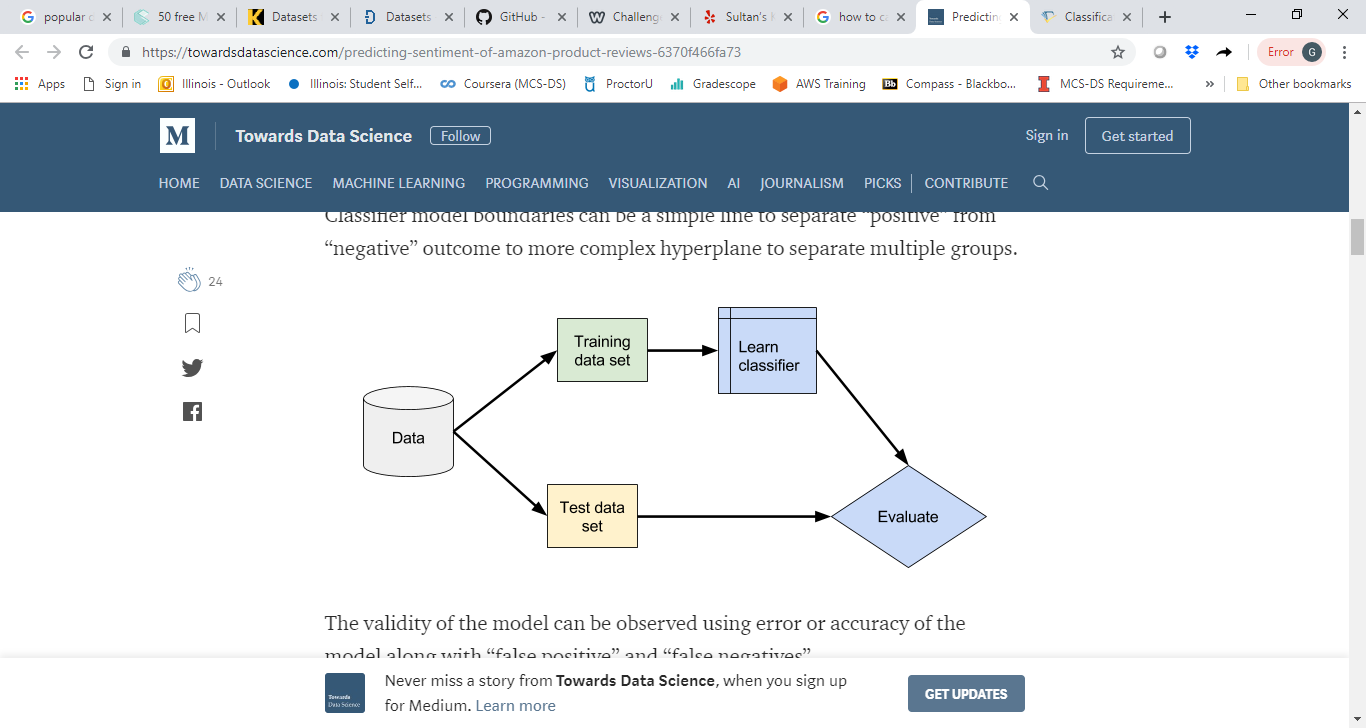
Classification is an instance of supervised learning. Training set has correctly identified observations. Classifier algorithms are used to create cluster/sets from the uncategorized unsupervised data based on similarity and/or distance from the training data set.

1. **Simple Statistics behind the analysis**

In simple classifier model, a simple count of positive and negative data points will define overall positive or negative sets. There is a problem with this. For example, in the case of words in sentences, “Great” and “Good” both are positive words. But “Great” has a higher impact than “Good”. We need to train our model to weigh identified data points.



Classifier model boundaries can be a simple line to separate “positive” from “negative” outcome to more complex hyperplane to separate multiple groups.



The validity of the model can be observed using error or accuracy of the model along with “false positive” and “false negatives”.

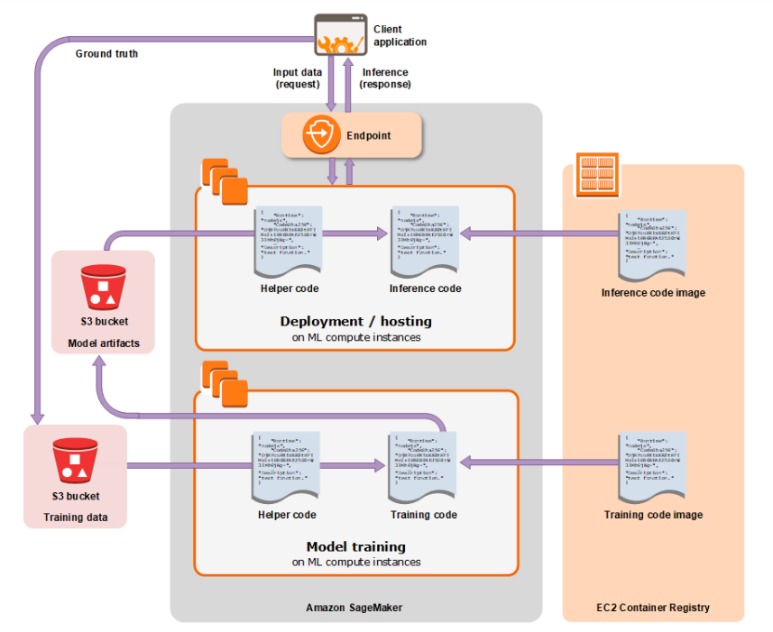
Error = Number of errors / Total number of data points

Accuracy = Number of corrected prediction / Total number of data points

1. **Amazon SageMaker and Sklearn – Technologies**

Amazon SageMaker is a fully managed machine learning service by Amazon hosted on AWS (Amazon Web Service) which covers the entire machine learning workflow to label and prepare data, choose an algorithm, train it and optimize it for deployment.

The system can store data in Amazon S3, use pandas for data reading, writing and management and use Sklearn toolkit for training model and real time predictions. Amazon SageMaker will create the test and predict pipeline with an end point to be used by external system for real time predictions. The high-level architecture diagram can be depicted as shown below:



We can easily expand training the dataset at scale in SageMaker's distributed, managed training environment. To use SageMaker's one of the pre-built container, we'll need to wrap our code into a Python script. There's a great deal of flexibility in using SageMaker's pre-built containers, and detailed documentation can be found [here](https://github.com/aws/sagemaker-python-sdk#mxnet-sagemaker-estimators).

Any larger dataset in Amazon S3 can be picked up by SageMaker training. This is perfect for use cases like periodic re-training, expanding to a larger dataset, or moving production workloads to larger hardware. A simple binary classifier model can be trained with the linear learner default settings and an Amazon SageMaker estimator can be created. By simply changing values such as train\_instance\_count and train\_instance\_type, one can easily change the size and number of instances to run on, which scales and distributes the training. Linear learner will stop training automatically after the model has converged. SageMaker has a number of hyperparameters that we can tune to improve model performance.

1. **Conclusion**

Using Amazon SageMaker, along with Sklearn toolkit, one can achieve state-of-the-art results on a real-world unstructured text classification task. In addition to the potential benefits that Deep Learning on SageMaker presents to our analysts and customers, it provides scalable solutions for companies, government agencies, and researchers interested in adding structure to large quantities of unstructured text. This could serve as the foundation for a recommender system in a variety of use cases. However, there are many ways in which it could be improved.

Beyond that, classifiers and recommenders are a very active area of research and techniques from active learning, reinforcement learning, segmentation, ensembling, and more should be investigated to deliver well-rounded recommendations.

1. **References**

<https://towardsdatascience.com/predicting-sentiment-of-amazon-product-reviews-6370f466fa73>

<https://blog.cambridgespark.com/50-free-machine-learning-datasets-sentiment-analysis-b9388f79c124>

<https://www.semanticscholar.org/paper/Lexicon-Based-Methods-for-Sentiment-Analysis-Taboada-Brooke/5250b0ceea33bcb01917777595e0f840fda6a49d>

<https://brandongaille.com/9-compelling-statistics-and-trends-about-yelp-reviews/>

<https://engineeringblog.yelp.com/2017/05/introducing-yelps-local-graph.html>

<https://github.com/awslabs/amazon-sagemaker-examples>